Conventions, Accuracy Metrics, Classification, Regression

Nipun Batra

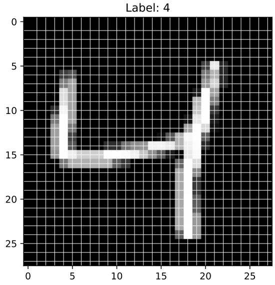
IIT Gandhinagar

July 29, 2025

Digit Recognition Problem

Let us work on the digit recognition problem.

Notebook: rule-based-vs-ml.html



Rule-based Approach for Digit Recognition

Maybe 4 can be thought of as: $| + — + | +$ another vertically down $|$

Rule-based Approach for Digit Recognition

Maybe 4 can be thought of as: $| + — + | +$ another vertically down $|$

Rule-based Approach for Digit Recognition

Maybe 4 can be thought of as: $| + — + | +$ another vertically down $|$

- The heights of each of the $|$ need to be similar within tolerance

Rule-based Approach for Digit Recognition

Maybe 4 can be thought of as: $| + — + | +$ another vertically down $|$

- The heights of each of the $|$ need to be similar within tolerance

Rule-based Approach for Digit Recognition

Maybe 4 can be thought of as: $| + — + | +$ another vertically down $|$

- The heights of each of the $|$ need to be similar within tolerance
- Each of the $|$ can be slightly slanted. Similarly the horizontal line can be slanted.

Rule-based Approach for Digit Recognition

Maybe 4 can be thought of as: $| + — + | +$ another vertically down $|$

- The heights of each of the $|$ need to be similar within tolerance
- Each of the $|$ can be slightly slanted. Similarly the horizontal line can be slanted.

Rule-based Approach for Digit Recognition

Maybe 4 can be thought of as: $| + — + | +$ another vertically down $|$

- The heights of each of the $|$ need to be similar within tolerance
- Each of the $|$ can be slightly slanted. Similarly the horizontal line can be slanted.
- There can be some cases of 4 where the first $|$ is at 45 degrees

Rule-based Approach for Digit Recognition

Maybe 4 can be thought of as: $| + — + | +$ another vertically down $|$

- The heights of each of the $|$ need to be similar within tolerance
- Each of the $|$ can be slightly slanted. Similarly the horizontal line can be slanted.
- There can be some cases of 4 where the first $|$ is at 45 degrees

Rule-based Approach for Digit Recognition

Maybe 4 can be thought of as: $| + — + | +$ another vertically down $|$

- The heights of each of the $|$ need to be similar within tolerance
- Each of the $|$ can be slightly slanted. Similarly the horizontal line can be slanted.
- There can be some cases of 4 where the first $|$ is at 45 degrees
- There can be some cases of 4 where the width of each stroke is different

Apple Quality Features

- Size

Apple Quality Features

- Size

Apple Quality Features

- Size

Apple Quality Features

- Size
- Colour

Apple Quality Features

- Size
- Colour

Apple Quality Features

- Size
- Colour
- Texture

Should We Include Sample Numbers?

Answer: Usually no! Sample numbers are typically arbitrary identifiers and not meaningful features. Let us remove it.

Should We Include Sample Numbers?

Answer: Usually no! Sample numbers are typically arbitrary identifiers and not meaningful features. Let us remove it.
Let us modify our data table for now.

Colour	**Size**	**Texture**	**Condition**
Orange	Small	Smooth	Good
Red	Small	Rough	Good
Orange	Medium	Smooth	Bad
Yellow	Large	Smooth	Bad

Training Set Components

The training set consists of two parts:

Training Set Components

The training set consists of two parts:

Training Set Components

The training set consists of two parts:

1. Features (Input Variables)

Training Set Components

The training set consists of two parts:

1. Features (Input Variables)

Training Set Components

The training set consists of two parts:

1. Features (Input Variables)
2. Output or Response Variable

Dataset Notation

We call this matrix as $\mathcal{D}$, containing:

1. Feature matrix ($\mathbf{X} \in \mathbb{R}^{n \times d}$) containing data of n samples each of which is d dimensional.

Dataset Notation

We call this matrix as $\mathcal{D}$, containing:

1. Feature matrix ($\mathbf{X} \in \mathbb{R}^{n \times d}$) containing data of n samples each of which is d dimensional.
2. Output vector ($\mathbf{y} \in \mathbb{R}^{n}$) containing output variable for n samples.

Dataset Example

Example (after encoding): $\mathbf{x}_1 = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$ (Orange=1, Small=0, Smooth=1)

Dataset Example

Example (after encoding): $\mathbf{x}_1 = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$ (Orange=1, Small=0, Smooth=1)

Dataset Example

Example (after encoding): $\mathbf{x}_1 = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$ (Orange=1, Small=0, Smooth=1)

- Complete dataset: $\mathcal{D} = \{(\mathbf{x}_i^\top, y_i)\}_{i=1}^n$

Machine Learning Goal

Learn f: Condition $= f(\text{colour, size, texture})$

Machine Learning Goal

Learn f: Condition $= f(\text{colour, size, texture})$

Machine Learning Goal

Learn f: Condition $= f(\text{colour, size, texture})$

1. From Training Dataset

Machine Learning Goal

Learn f: Condition $= f$(colour, size, texture)

1. From Training Dataset

Machine Learning Goal

Learn f: Condition $= f$(colour, size, texture)

1. From Training Dataset
2. To Predict the condition for the Testing set

Colour	**Size**	**Texture**	**Condition**
Orange	Small	Smooth	Good
Red	Small	Rough	Good
Orange	Medium	Smooth	Bad
Yellow	Large	Smooth	Bad
Red	Large	Rough	?
Orange	Large	Rough	?

Can We Judge Performance Only on Test Set?

A: Ideally, no!

Can We Judge Performance Only on Test Set?

A: Ideally, no!

Can We Judge Performance Only on Test Set?

A: Ideally, no!

- Ideally - we want to predict "well" on all possible inputs. But, can we test that?

Can We Judge Performance Only on Test Set?

A: Ideally, no!

- Ideally - we want to predict "well" on all possible inputs. But, can we test that?

Can We Judge Performance Only on Test Set?

A: Ideally, no!

- Ideally - we want to predict "well" on all possible inputs. But, can we test that?
- No! Since, the test set is only a sample from all possible inputs.

Training vs Test Sets

Both the training set and the test set are samples drawn from the hidden true distribution (also sometimes called population)

Training vs Test Sets

Both the training set and the test set are samples drawn from the hidden true distribution (also sometimes called population)

More discussion later once we study bias and variance

Energy Consumption Example

- # People (More people $\implies$ More Energy)

Energy Consumption Example

- # People (More people $\implies$ More Energy)

Energy Consumption Example

- # People (More people $\implies$ More Energy)

Energy Consumption Example

- # People (More people $\implies$ More Energy)
- Temperature (Higher Temp. $\implies$ Higher Energy)

Energy Consumption Example

- # People (More people $\implies$ More Energy)
- Temperature (Higher Temp. $\implies$ Higher Energy)

# People	**Temp (C)**	**Energy (kWh)**
4000	30	30
4200	30	32
4200	35	40
3000	20	?
1000	45	?

Classification vs Regression

- Classification

Classification vs Regression

- Classification

Classification vs Regression

- Classification

Classification vs Regression

- Classification
  - Output variable is discrete

Classification vs Regression

- Classification
  - Output variable is discrete

Classification vs Regression

- Classification
  - Output variable is discrete
  - i.e. $y_i \in \{1, 2, \ldots, k\}$ where k is number of classes

Classification vs Regression

- Classification
  - Output variable is discrete
  - i.e. $y_i \in \{1, 2, \ldots, k\}$ where k is number of classes

Classification vs Regression

- Classification
  - Output variable is discrete
  - i.e. $y_i \in \{1, 2, \ldots, k\}$ where k is number of classes
  - Examples - Predicting:

Classification vs Regression

- Classification
  - Output variable is discrete
  - i.e. $y_i \in \{1, 2, \dots, k\}$ where k is number of classes
  - Examples - Predicting:

Classification vs Regression

- Classification
  - Output variable is discrete
  - i.e. $y_i \in \{1, 2, \ldots, k\}$ where k is number of classes
  - Examples - Predicting:
    - Will I get a loan? (Yes, No)

Classification vs Regression

- Classification
  - Output variable is discrete
  - i.e. $y_i \in \{1, 2, \ldots, k\}$ where k is number of classes
  - Examples - Predicting:
    - Will I get a loan? (Yes, No)

Classification vs Regression

- Classification
  - Output variable is discrete
  - i.e. $y_i \in \{1, 2, \ldots, k\}$ where k is number of classes
  - Examples - Predicting:
    - Will I get a loan? (Yes, No)
    - What is the quality of fruit? (Good, Bad)

Classification vs Regression

- Classification
  - Output variable is discrete
  - i.e. $y_i \in \{1, 2, \ldots, k\}$ where k is number of classes
  - Examples - Predicting:
    - Will I get a loan? (Yes, No)
    - What is the quality of fruit? (Good, Bad)

Classification vs Regression

- Classification
  - Output variable is discrete
  - i.e. $y_i \in \{1, 2, \ldots, k\}$ where k is number of classes
  - Examples - Predicting:
    - Will I get a loan? (Yes, No)
    - What is the quality of fruit? (Good, Bad)
- Regression

Classification vs Regression

- Classification
  - Output variable is discrete
  - i.e. $y_i \in \{1, 2, \ldots, k\}$ where k is number of classes
  - Examples - Predicting:
    - Will I get a loan? (Yes, No)
    - What is the quality of fruit? (Good, Bad)
- Regression

Classification vs Regression

- Classification
  - Output variable is discrete
  - i.e. $y_i \in \{1, 2, \ldots, k\}$ where k is number of classes
  - Examples - Predicting:
    - Will I get a loan? (Yes, No)
    - What is the quality of fruit? (Good, Bad)
- Regression
  - Output variable is continuous

Classification vs Regression

- Classification
  - Output variable is discrete
  - i.e. $y_i \in \{1, 2, \ldots, k\}$ where k is number of classes
  - Examples - Predicting:
    - Will I get a loan? (Yes, No)
    - What is the quality of fruit? (Good, Bad)
- Regression
  - Output variable is continuous

Classification vs Regression

- Classification
  - Output variable is discrete
  - i.e. $y_i \in \{1, 2, \ldots, k\}$ where k is number of classes
  - Examples - Predicting:
    - Will I get a loan? (Yes, No)
    - What is the quality of fruit? (Good, Bad)
- Regression
  - Output variable is continuous
  - i.e. $y_i \in \mathbb{R}$

Classification vs Regression

- Classification
  - Output variable is discrete
  - i.e. $y_i \in \{1, 2, \ldots, k\}$ where k is number of classes
  - Examples - Predicting:
    - Will I get a loan? (Yes, No)
    - What is the quality of fruit? (Good, Bad)
- Regression
  - Output variable is continuous
  - i.e. $y_i \in \mathbb{R}$

Classification vs Regression

- Classification
  - Output variable is discrete
  - i.e. $y_i \in \{1, 2, \ldots, k\}$ where k is number of classes
  - Examples - Predicting:
    - Will I get a loan? (Yes, No)
    - What is the quality of fruit? (Good, Bad)
- Regression
  - Output variable is continuous
  - i.e. $y_i \in \mathbb{R}$
  - Examples - Predicting:

Classification vs Regression

- Classification
  - Output variable is discrete
  - i.e. $y_i \in \{1, 2, \ldots, k\}$ where k is number of classes
  - Examples - Predicting:
    - Will I get a loan? (Yes, No)
    - What is the quality of fruit? (Good, Bad)
- Regression
  - Output variable is continuous
  - i.e. $y_i \in \mathbb{R}$
  - Examples - Predicting:

Classification vs Regression

- Classification
  - Output variable is discrete
  - i.e. $y_i \in \{1, 2, \ldots, k\}$ where k is number of classes
  - Examples - Predicting:
    - Will I get a loan? (Yes, No)
    - What is the quality of fruit? (Good, Bad)
- Regression
  - Output variable is continuous
  - i.e. $y_i \in \mathbb{R}$
  - Examples - Predicting:
    - How much energy will campus consume?

Classification vs Regression

- Classification
  - Output variable is discrete
  - i.e. $y_i \in \{1, 2, \ldots, k\}$ where k is number of classes
  - Examples - Predicting:
    - Will I get a loan? (Yes, No)
    - What is the quality of fruit? (Good, Bad)
- Regression
  - Output variable is continuous
  - i.e. $y_i \in \mathbb{R}$
  - Examples - Predicting:
    - How much energy will campus consume?

Classification vs Regression

- Classification
  - Output variable is discrete
  - i.e. $y_i \in \{1, 2, \ldots, k\}$ where k is number of classes
  - Examples - Predicting:
    - Will I get a loan? (Yes, No)
    - What is the quality of fruit? (Good, Bad)
- Regression
  - Output variable is continuous
  - i.e. $y_i \in \mathbb{R}$
  - Examples - Predicting:
    - How much energy will campus consume?
    - How much rainfall will fall?

Accuracy Calculation

$$\text{Accuracy} = \frac{|\{i : y_i = \hat{y}_i\}|}{n} = \frac{3}{5} = 0.6$$

Accuracy Notation

- **Set cardinality notation:** $|\{i : y_i = \hat{y}_i\}|$

Accuracy Notation

- **Set cardinality notation:** $|\{i : y_i = \hat{y}_i\}|$
  - Reads as: "Number of indices i such that $y_i = \hat{y}_i$"

Accuracy Notation

- **Set cardinality notation:** $|\{i : y_i = \hat{y}_i\}|$
  - Reads as: "Number of indices i such that $y_i = \hat{y}_i$"
  - Counts how many samples satisfy the condition

Accuracy Notation

- **Set cardinality notation:** $|\{i : y_i = \hat{y}_i\}|$
  - Reads as: "Number of indices i such that $y_i = \hat{y}_i$"
  - Counts how many samples satisfy the condition

Accuracy Notation

- **Set cardinality notation:** $|\{i : y_i = \hat{y}_i\}|$
  - Reads as: "Number of indices i such that $y_i = \hat{y}_i$"
  - Counts how many samples satisfy the condition

Accuracy Notation

- **Set cardinality notation:** $|\{i : y_i = \hat{y}_i\}|$
  - Reads as: "Number of indices i such that $y_i = \hat{y}_i$"
  - Counts how many samples satisfy the condition
- **Alternative: Indicator function notation**

$$\text{Accuracy} = \frac{\sum_{i=1}^{n} \mathbf{1}[y_i = \hat{y}_i]}{n}$$

$$\text{where } \mathbf{1}[\text{condition}] = \begin{cases} 1 & \text{if condition is true} \\ 0 & \text{if condition is false} \end{cases}$$

Accuracy Notation

- **Set cardinality notation:** $|\{i : y_i = \hat{y}_i\}|$
  - Reads as: "Number of indices i such that $y_i = \hat{y}_i$"
  - Counts how many samples satisfy the condition
- **Alternative: Indicator function notation**

$$\text{Accuracy} = \frac{\sum_{i=1}^{n} \mathbf{1}[y_i = \hat{y}_i]}{n}$$

$$\text{where } \mathbf{1}[\text{condition}] = \begin{cases} 1 & \text{if condition is true} \\ 0 & \text{if condition is false} \end{cases}$$

Accuracy Notation

- **Set cardinality notation:** $|\{i : y_i = \hat{y}_i\}|$
  - Reads as: "Number of indices i such that $y_i = \hat{y}_i$"
  - Counts how many samples satisfy the condition
- **Alternative: Indicator function notation**

$$\text{Accuracy} = \frac{\sum_{i=1}^{n} \mathbf{1}[y_i = \hat{y}_i]}{n}$$

$$\text{where } \mathbf{1}[\text{condition}] = \begin{cases} 1 & \text{if condition is true} \\ 0 & \text{if condition is false} \end{cases}$$

- Both notations are mathematically equivalent and commonly used in ML literature

When Precision/Recall Matter

Cases for this:

- Cancer Screening

When Precision/Recall Matter

Cases for this:

- Cancer Screening
- Planet Detection

Precision Metric

$$\text{Precision} = \frac{|\{i : y_i = \hat{y}_i = \text{Good}\}|}{|\{i : \hat{y}_i = \text{Good}\}|} = \frac{2}{4} = 0.5$$

"the fraction of relevant instances among the retrieved instances", i.e. "out of the number of times we predict Good, how many times is the condition actually Good"

Accuracy vs Precision/Recall

$$\begin{aligned}
\text{Accuracy} &= \frac{98}{100} = 0.98 \\
\text{Recall} &= \frac{0}{1} = 0 \\
\text{Precision} &= \frac{0}{1} = 0
\end{aligned}$$

Confusion Matrix

		Ground Truth	
		Yes	No
Predicted	Yes	True Positive	False Positive
	No	False Negative	True Negative

Example Metrics

$$
\begin{array}{c} \\ \text{Pred Positive} \\ \text{Pred Negative} \end{array}
\begin{array}{c}
\begin{array}{cc} \text{G.T. Positive} & \text{G.T. Negative} \end{array} \\
\begin{pmatrix} 0 & 1 \\ 1 & 98 \end{pmatrix}
\end{array}
$$

Example Metrics

$$
\begin{array}{r c}
 & \begin{array}{cc} \text{G.T. Positive} & \text{G.T. Negative} \end{array} \\
\begin{array}{r} \text{Pred Positive} \\ \text{Pred Negative} \end{array} & \begin{pmatrix} 0 & 1 \\ 1 & 98 \end{pmatrix}
\end{array}
$$

$$
\begin{array}{r c}
 & \begin{array}{cc} \text{G.T. Positive} & \text{G.T. Negative} \end{array} \\
\begin{array}{r} \text{Pred Positive} \\ \text{Pred Negative} \end{array} & \begin{pmatrix} \text{True Positive} & \text{False Positive} \\ \text{True Negative} & \text{False Negative} \end{pmatrix}
\end{array}
$$

Example Metrics

$$\begin{array}{c} \\ \text{Pred Positive} \\ \text{Pred Negative} \end{array} \begin{array}{c} \begin{array}{cc} \text{G.T. Positive} & \text{G.T. Negative} \end{array} \\ \begin{pmatrix} 0 & 1 \\ 1 & 98 \end{pmatrix} \end{array}$$

$$\begin{array}{c} \\ \text{Pred Positive} \\ \text{Pred Negative} \end{array} \begin{array}{c} \begin{array}{cc} \text{G.T. Positive} & \text{G.T. Negative} \end{array} \\ \begin{pmatrix} \text{True Positive} & \text{False Positive} \\ \text{True Negative} & \text{False Negative} \end{pmatrix} \end{array}$$

$$\text{Recall} = \frac{\text{T.P}}{\text{T.P} + \text{F.P}}$$

$$\text{Precision} = \frac{\text{T.P}}{\text{T.P} + \text{F.N}}$$

Mean Error Issues

Is there any downside with using mean error?

Mean Error Issues

Is there any downside with using mean error? Errors can get cancelled out

Pop Quiz

Quick Quiz 1

Which metrics should you use for imbalanced datasets?

1. Accuracy only

Answer: c) Precision, recall, and F1-score give a more complete picture!

Pop Quiz

Quick Quiz 1

Which metrics should you use for imbalanced datasets?

1. Accuracy only
2. Mean squared error

Answer: c) Precision, recall, and F1-score give a more complete picture!

Pop Quiz

Quick Quiz 1

Which metrics should you use for imbalanced datasets?

1. Accuracy only
2. Mean squared error
3. Precision, recall, and F1-score

Answer: c) Precision, recall, and F1-score give a more complete picture!

Key Takeaways

- **ML vs Traditional Programming:** ML learns rules from data, traditional programming uses predefined rules

Key Takeaways

- **ML vs Traditional Programming:** ML learns rules from data, traditional programming uses predefined rules

Key Takeaways

- **ML vs Traditional Programming:** ML learns rules from data, traditional programming uses predefined rules
- **Features matter:** Choose meaningful features, avoid arbitrary identifiers

Key Takeaways

- **ML vs Traditional Programming:** ML learns rules from data, traditional programming uses predefined rules
- **Features matter:** Choose meaningful features, avoid arbitrary identifiers

Key Takeaways

- **ML vs Traditional Programming:** ML learns rules from data, traditional programming uses predefined rules
- **Features matter:** Choose meaningful features, avoid arbitrary identifiers
- **Classification vs Regression:** Discrete outputs vs continuous outputs

Key Takeaways

- **ML vs Traditional Programming:** ML learns rules from data, traditional programming uses predefined rules
- **Features matter:** Choose meaningful features, avoid arbitrary identifiers
- **Classification vs Regression:** Discrete outputs vs continuous outputs

Key Takeaways

- **ML vs Traditional Programming:** ML learns rules from data, traditional programming uses predefined rules
- **Features matter:** Choose meaningful features, avoid arbitrary identifiers
- **Classification vs Regression:** Discrete outputs vs continuous outputs
- **Accuracy isn't everything:** For imbalanced data, use precision, recall, F1-score

Key Takeaways

- **ML vs Traditional Programming:** ML learns rules from data, traditional programming uses predefined rules
- **Features matter:** Choose meaningful features, avoid arbitrary identifiers
- **Classification vs Regression:** Discrete outputs vs continuous outputs
- **Accuracy isn't everything:** For imbalanced data, use precision, recall, F1-score

Key Takeaways

- **ML vs Traditional Programming:** ML learns rules from data, traditional programming uses predefined rules
- **Features matter:** Choose meaningful features, avoid arbitrary identifiers
- **Classification vs Regression:** Discrete outputs vs continuous outputs
- **Accuracy isn't everything:** For imbalanced data, use precision, recall, F1-score
- **Visualization is crucial:** Always plot your data (Anscombe's Quartet lesson)

Key Takeaways

- **ML vs Traditional Programming:** ML learns rules from data, traditional programming uses predefined rules
- **Features matter:** Choose meaningful features, avoid arbitrary identifiers
- **Classification vs Regression:** Discrete outputs vs continuous outputs
- **Accuracy isn't everything:** For imbalanced data, use precision, recall, F1-score
- **Visualization is crucial:** Always plot your data (Anscombe's Quartet lesson)

Key Takeaways

- **ML vs Traditional Programming:** ML learns rules from data, traditional programming uses predefined rules
- **Features matter:** Choose meaningful features, avoid arbitrary identifiers
- **Classification vs Regression:** Discrete outputs vs continuous outputs
- **Accuracy isn't everything:** For imbalanced data, use precision, recall, F1-score
- **Visualization is crucial:** Always plot your data (Anscombe's Quartet lesson)
- **Use baselines:** Simple baseline models help validate your approach

Summary: Evaluation Metrics

Task	Common Metrics	When to Use
Classification	Accuracy, Precision, Recall, F1 Confusion Matrix	Balanced/Imbalanced Multi-class problems
Regression	MSE, RMSE, MAE Mean Error	Continuous prediction Check for bias

Remember: Choose metrics based on your problem's characteristics and business requirements!